(19) BUNDESREPUBLIK DEUTSCHLAND

DEUTSCHES PATENTAMT

(12) Übersetzung der europäischen Patentschrift

(87) EP 0 626 664 B1

(10) DE 694 00 549 T 2

(51) Int. Cl.[6]: G 07 F 7/10
G 06 F 9/44
H 04 M 17/00

(21) Deutsches Aktenzeichen: 694 00 549.5
(86) Europäisches Aktenzeichen: 94 400 900.0
(86) Europäischer Anmeldetag: 26. 4. 94
(87) Erstveröffentlichung durch das EPA: 30. 11. 94
(87) Veröffentlichungstag der Patenterteilung beim EPA: 18. 9. 96
(47) Veröffentlichungstag im Patentblatt: 30. 1. 97

(30) Unionspriorität: (32) (33) (31)
28.04.93 FR 9305023

(73) Patentinhaber:
Gemplus Card International, Gemenos, FR

(74) Vertreter:
Beetz und Kollegen, 80538 München

(84) Benannte Vertragstaaten:
DE, ES, GB, IT, NL

(72) Erfinder:
Martineau, Philippe, Cabinet BALLOT-SCHMIT, F-75116 Paris, FR

(54) IC-Karten-Übertragungssystem

DE 694 00 549 T 2

DE 694 00 549 T 2

Die vorliegende Erfindung bezieht sich auf ein Kommunikationssystem zwischen einem Zentralserver und einem Nutzer über eine Schnittstelle. Sie zielt insbesondere auf die Bereiche, wo die Funktion der Schnittstelle bestimmt wird durch einen elektronischen Mikrochip vom Typ einer Münze, die einen Chip trägt, Chipkarte oder dgl., die zeitweilig herausnehmbar oder definitiv in die Schnittstelle eingeführt wird. Man findet Anwendungen dieses Systems bei tragbaren Telefonen, die frei verfügbar sind und bei denen man z.B. die Benutzung des Telefons durch denjenigen, der den entsprechenden Schaltkreis nicht eingeführt hat, verhindern will. Sie betrifft außerdem alle Anschlüsse, bei denen direkte Bezahlung mittels Chipkarten erfolgt, insbesondere beim Abheben von Bankkonten.

Das Problem, das bei diesem Systemtyp auftritt, hängt mit der Definition der Schnittstelle zusammen. Tatsächlich erfordern in einem gegebenen Zeitpunkt die bekannten Sicherheitsanforderungen und/oder der Bedienungskomfort das Aufrufen von Prozeduren über die Schnittstelle. Diese Prozeduren werden weiterentwickelt. Dies hat zur Konsequenz oder zum Nachteil, daß die veralteten Schnittstellen je nach dieser Entwicklung aktualisiert werden müssen.

Es könnte trotzdem angestrebt werden, die Betriebssysteme der Schnittstellen in ähnlicher Weise wie die Systeme auf der Basis der Mikrorechner auf den neuesten Stand zu bringen. Jedoch hätte dieses Vorgehen den Nachteil, daß viele Aktualisierungen durchgeführt werden müßten. Darüber hinaus

ist der Besitzer oder der Verwalter der Schnittstelle nicht der Nutzer dieser Schnittstelle: derjenige, der den elektronischen Schaltkreis zur fraglichen Autorisierung besitzt. Zum Beispiel ist bei einem Zahlungsanschluß für eine Chipkarte in einem Restaurant der Besitzer der Schnittstelle der Restaurantbetreiber und also der Nutzer, der Zahlende der Inhaber der Chipkarte. Dasselbe ist es bei Systemen von tragbaren Telefonen, insbesondere in dem Bereich der Autovermietung, wo das tragbare Telefon (also die Schnittstelle) der Autovermietungsfirma (oder einer Vertragsgesellschaft eines Telefonnetzes) gehört und wo die Chipkarte oder der betreffende elektronische Schaltkreis Eigentum des Nutzers ist. In diesem Fall kann der Unterschied zwischen einer Schnittstelle und einer anderen aufgrund einer Aktualisierung für den Nutzer die Benutzung sehr kompliziert machen.

Es ist ein Kompromiß notwendig zwischen dem Bedürfnis des Nutzers, der daran interessiert ist, mit seiner Chipkarte über die letzten Neuerungen zu verfügen, und dem Bedürfnis des Verwalters der Schnittstelle, der aus Kostengründen dazu neigt, die Zahl der Aktualisierungsvorgänge bei den von ihm verwalteten Schnittstellen einzuschränken.

Ein praktisches Beispiel ist der Bedarf, der im Bereich der tragbaren Telefone empfunden wird, wo der Verwalter der Schnittstelle oder eben der Besitzer des Servers die vorbezahlten Einheiten auf der Chipkarte des Nutzers abrechnen können möchte, in etwa wie dies bei den öffentlichen Telefonen erfolgt. Mit anderen Worten, die tragbaren Telefonapparate heute, die Schnittstellen, sind in ihrem Betriebssystem nicht mit einem Befehlssatz ausgestattet, der die Ab-

rechnung erlaubt. Dies wird später der Fall sein. Das zu lösende Problem in diesem besonderen Fall ist es daher, mit einer Schnittstelle, die nicht dafür ausgerüstet ist, die Abrechnungsimpulse, die von einem Telefonserver abgeschickt würden, auf einer Chipkarte abzurechnen. Man erkennt gut die Schwierigkeit bei diesem Problem.

EP-A-0 292 248 beschreibt ein System für die Fernübertragung von Befehlen auf eine Chipkarte. In "L'écho des recherches" Nr. 139, 1990, S. 13-20, X P 386 290 wird von P. Jolie und anderen die Benutzung einer Chipkarte in einem Funktelefon beschrieben. Keine dieser Offenbarungen ist jedoch verwendbar, um im Verlauf einer Verbindung die Ausführung eines Befehls zu ermöglichen, der nicht schon in der Schnittstelle oder auf der Chipkarte vorhanden ist.

Der Erfindung liegt die Aufgabe zugrunde, eine Lösung für dieses Problem zu finden und eine sehr viel größere Flexibilität bei der Nutzung der Schnittstellen oder Nutzeranschlüsse zu gewährleisten. Mehrere Lösungen sind vorstellbar, man könnte insbesondere einen "universellen" Mechanismus definieren, der die vom Server her transparente Verwaltung der Anschlüsse erlaubt. Dazu müßte jeder existierende Anschluß mit diesem neuen Mechanismus ausgestattet werden. Es ist daher eine wenig realistische Lösung. Die erfindungsgemäß vorgeschlagene Lösung erfordert keine Modifizierung des Anschlusses und nutzt nur die bereits vorhandenen Ressourcen desselben aus. Diese Ressourcen sind entweder vom Server her verfügbar oder Kartenbefehle (eingespeist in den Anschluß), gesteuert vom Server her, die die Weiterleitung von Daten über den Anschluß auf die Karte erlauben. Im übrigen erhalten erfindungsgemäß die Anschlüsse ein Be-

triebssystem, das einfacher ist und das schließlich nicht einmal mehr verändert werden muß.

Die der Erfindung zugrunde liegende Idee ist die Verwendung eines minimalen Befehlssatzes (Instruktionssatzes)oder selbst einer existierenden Prozedur neben dem existierenden Befehlssatz in einem Betriebssystem von Schnittstellen. Mit dem Satz oder der Prozedur erfolgt das Zeigen auf einen zusätzlichen Satz oder ausgedehnten Satz in einer reservierten Zone im Speicher des Chips. Dieser zusätzliche Befehl (oder Instruktion) wird dann entweder durch den Mikroprozessor auf der Chipkarte oder durch den Mikroprozessor der Schnittstelle ausgeführt, wobei letztere auf die bezeichnete Peripherie einwirken. Sie bewirken z.B. die Anzeige von Informationen auf dem Schirm der Schnittstelle, eine Abrechnung der vorgezahlten Einheiten im Speicher der Chipkarte, ein Abschicken der relativen Bilanz der Chipkarte und/oder den laufenden Austausch (insbesondere um Rechnungen über die Dauer der Übertragung zu erstellen) mit dem Server, oder eine Annullierung der Chipkarte, etc...

Mit einer existierenden reservierten Zone, deren Platz und folglich Benutzung bekannt sind, kann man so mit dem minimalen Befehlssatz des Betriebssystems der Schnittstelle die Eigenschaften eines Befehls laden. Die Eigenschaften dieses Befehls sind entweder der Befehlscode des Befehls selbst oder eine Adresse des Befehls in einem Speicher des Chips. Über diese reservierte Zone hinaus umfaßt der Mikroprozessor der Chipkarte einen Ausführungsautomatismus des Befehls, dessen Eigenschaften in der reservierten Zone geladen sind. Wenn man die Nutzbarkeit eines Systems verbessern will, schickt man an die Chipkarte die Eigenschaften des

auszuführenden Befehls vom Server ab. Diese werden naturgemäß über die Schnittstelle übertragen, die in der Praxis eine Speicherzone im Speicher der Chipkarte auszuwählen und zu beschreiben (dies ist ein Minimum) weiß und dort die Eigenschaften dieses Befehls hineinschreiben kann. Im folgenden fährt der Mikroprozessor der Chipkarte mit seinem Automatismus fort und leitet die Ausführung dieses Befehls ein.

Bei einer Weiterentwicklung lädt man statt den Befehl vom Server aus abzuschicken die zusätzlichen Befehle in die auf den neuesten Stand gebrachten Karten in die zusätzlichen Befehlssätze. In diesem Fall bestehen die Befehlseigenschaften, die vom Server abgeschickt werden müssen, nur aus einem Zeiger, der denjenigen der zusätzlichen Befehle anzeigt, den man ausgeführt haben möchte.

Bei einer Variante verwendet man eine existierende Prozedur zur Verwaltung der Karte, die mit dem Abschicken von Daten assoziiert ist. Man kennzeichnet eine Dateneinheit und verifiziert beim Empfang auf der Karte, daß die übertragene Dateneinheit diese gekennzeichnete Form hat. Ggf. führt man einen Befehl des erweiterten Satzes aus, der dann dieser gekennzeichneten Dateneinheit entspricht.

Die Erfindung stellt sich daher die Aufgabe, ein Kommunikationssystem anzugeben, das umfaßt:

einen Zentralserver, einen elektronischen Chip auf einem Chipträger und eine Schnittstelle für die Kommunikation zwischen Zentralserver, diesem Chip und evtl. einem Nutzer,

in der Schnittstelle einen Mikroprozessor und einen Programmspeicher, der mit einem begrenzten Satz an Instruktionen oder Prozeduren zur Kommunikation mit dem Chip ausgerüstet ist, und

in dem Chip einen Mikroprozessor und einen Programmspeicher, der gleichfalls mit einem entsprechenden begrenzten Satz an Instruktionen oder Prozeduren ausgestattet ist,

dadurch gekennzeichnet, daß das Kommunikationssystem umfaßt:

Vorrichtungen, die dazu dienen, aufgrund einer vom Zentralserver ausgegebenen Meldung im Verlauf einer Nutzungsperiode in einer reservierten Speicherzone des Chips mit den begrenzten Sätzen an Instruktionen oder Prozeduren des Chips und/oder der Schnittstelle die Eigenschaften einer Instruktion auszuwählen oder zu schreiben, die sich von denen der begrenzten Sätze oder Prozeduren unterscheidet, und

in dem Chip einen Ausführungsautomatismus, der dazu dient, im Verlauf dieser Periode diese unterschiedliche Instruktion nach Auswahl oder nach Schreiben ihrer Eigenschaften in diese reservierte Speicherzone auszuführen.

Die Erfindung wird deutlich im Verlauf der folgenden Beschreibung unter Heranziehung der beigefügten Figuren. Letztere sind lediglich Andeutungen, auf die die Erfindung keineswegs beschränkt ist. Die Figuren zeigen:

Fig. 1 eine allgemeine Darstellung eines erfindungsgemäßen Kommunikationssystems;

Fig. 2 eine schematische Darstellung des Ablaufs des Automatismus bei Ausführung der Instruktion aus dem zusätzlichen Satz;

Fig. 3 Mikroprogramme, die in einer Chipkarte aufgerufen werden, welche in dem erfindungsgemäßen Kommunikationssystem eingesetzt wird.

Fig. 1 stellt ein erfindungsgemäßes Kommunikationssystem mit einem Zentralserver 1, einem elektronischen Chip 2 auf z.B. einem Chipträger 3, z.B. einer Chipkarte oder einer Chipmünze oder dgl., insbesondere einem Träger mit Anschlüssen für eine integrierte Schaltung in einem DIP- oder CMS-Gehäuse. Dieses System umfaßt außerdem eine Kommunikationsschnittstelle 4 zwischen dem Zentralserver 1, diesem Chip 2 und evtl. einem Nutzer, der die Eingabetasten 5 der Schnittstelle 4 betätigen kann. Der Server 1 ist das System, das die Applikation beherbergt und sie den unterschiedlichen Schnittstellen und Nutzern zur Verfügung stellt. Der Server 1 schickt die Informationen an die Schnittstellen sowie die Karte über eine Übertragungsverbindung 6. Es kann jedoch auch eine Direktverbindung sein. Die Übertragungsverbindung 6 kann außerdem ein Weg in einem Übertragungsnetz für Daten sein und unterschiedliche Formen haben: Verdrahtung, Funk oder andere. Dazu werden Übertragungsprotokoll und insbesondere Modems 7 und 8 für das Weiterleiten der Informationen verwendet. Die Informationen stellen die Daten, Adressen und/oder Befehls dar.

Die Schnittstelle 4 kann für eine Applikation z.B. vom Typ EFTPOS (Electronic Found Transfer Point of Sale) sein oder

einem Standard entsprechen, z.B. vom Typ PC oder Minitel. Die Rolle der Schnittstellen besteht im wesentlichen darin, einem Nutzer den Dialog mit dem Server 1 oder dem Chip 2 zu ermöglichen und/oder für den Server 1, den Chip 2 oder die Schnittstelle 4 bestimmte Informationen aufzusetzen und weiterzuleiten. Die Chipkarte 3 umfaßt für die Applikation spezifische und für den Inhaber spezifische Daten: den Nutzer. Sie verfügt über ein Betriebssystem, das ein Arsenal von Funktionen dank einem meistens spezifischen Befehlssatz bietet.

In der Schnittstelle 4 ist ein Mikroprozessor 9 installiert, der mit einem Bus 10 für Befehle, Daten und Adressen hauptsächlich mit einem Programmspeicher 11, einem Lesegerät für den Chipträger 12 und einem Eingangs/Ausgangsschaltkreis zum Server 1, z.B. dem Modem 8, verbunden ist. In dieser Minimalversion dient die Schnittstelle 4 nur dazu, die Verbindung mit der Chipkarte 3 sicherzustellen. Bei Bedarf ist der Bus 10 ebenfalls verbunden mit einem Anzeigeschirm 13, was die Darstellung der ausgeführten Operationen erlaubt. Vorzugsweise ist der Bus 10 ebenfalls mit Eingabetasten 5 verbunden, was das Einschreiten eines Nutzers ermöglicht. In diesem Fall ist die Schnittstelle eine Schnittstelle zwischen Bediener und der Chipkarte oder zwischen dem Server, Bediener und Chipkarte. Der Programmspeicher 11 beinhaltet einen begrenzten Befehlssatz, hier die Befehle SELECT, READ, UPDATE, STATUS, die nicht sehr zahlreich sind. In der Praxis sind nur die Befehle SELECT und UPDATE wichtig, die jeweils die Auswahl einer Speicherzone des Speichers des Chips der Chipkarte und/oder das Schreiben von irgend etwas erlauben. Jedoch aus Gründen des Komforts wird dieser eingeschränkte Satz ebenfalls einen Be-

fehl nur zum Lesen, READ, in einer Zone der Chipkarte und vorzugsweise evtl. einen Befehl STATUS umfassen, der den Zustand der internen Zähler oder Register der Chipkarte an den Server zu schicken und/oder sie auf dem Schirm 13 sichtbar zu machen erlaubt. Der Inhalt dieses Befehls bleibt noch zu definieren. Er könnte auch ein Joker-Befehl sein.

Der Chip 2 der Karte 3 umfaßt einen Mikroprozessor 14 und einen Programmspeicher 15 mit wenigstens einem eingeschränkten Befehlssatz 16. Der Satz 16 entspricht dem Satz 11: man findet in ihm dieselben Befehle wie im letzteren, d.h., die Befehle SELECT, READ, UPDATE und STATUS. Hier sind ebenfalls nur die Befehle SELECT und UPDATE für die Erfindung wichtig. Man wird bemerken, daß die eingeschränkten Sätze 11 und 16 etwas vollständiger als die Minimalsätze sein können. Der Chip 2 umfaßt außerdem einen Bus 17 vom selben Typ wie der Bus 10, über den der Mikroprozessor 14 verbunden ist mit dem Programmspeicher 15 und einer Eingangs/Ausgangsvorrichtung 18. Im Fall von Chipkarten umfaßt die Eingangs/Ausgangsvorrichtung 18 einen normalisierten Kontakt, dessen metallische Flächen dazu bestimmt sind, zu den Abgreifern 19 des Lesegeräts 12 Verbindung herzustellen.

Eine der wesentlichen Eigenschaften der Erfindung ist es, daß der Chip 2 eine reservierte Speicherzone 20 hat, die über den Bus 17 mit dem Mikroprozessor 14 verbunden ist. In dieser reservierten Zone 20 veranlaßt der Server 1 das Einschreiben der Eigenschaften eines Befehls, der sich von denen der beschränkten Sätze 11 und 16 unterscheidet. Er veranlaßt z.B. das Schreiben der Eigenschaften ein Befehls

BILAN (anders als STATUS), der dazu dient, über den Mikroprozessor 14 auf den Schirm 13 den Saldozustand der vorbezahlten und in einem angehängten Speicher 21 der Karte enthaltenen Einheiten abzuschicken. Dieser andere zusätzliche Befehl kann außerdem ein Befehl ALGO zum Ausführen eines Algorithmus zur Chiffrierung einer Meldung oder der Anerkennung der Chipkarte 3 oder des Anschlusses 4 durch den Mikroprozessor 14 sein. Dieser andere Befehl kann außerdem ein Befehl CHANGE zum Ändern des Wertes der vorbezahlten Einheiten sein, um gewisse Einheiten 22 im Verlauf der Kommunikationsdauer zu entwerten. Er kann außerdem ein Befehl INVAL zur Annullierung der Karte sein oder jeder andere Befehl, dessen Notwendigkeit sich nach und nach im Verlauf der Entwicklungen der Applikationen selbst herausstellt.

Anstatt die Eigenschaften eines Befehls, der sich von denen des beschränkten Satzes unterscheidet, in der Zone 20 des reservierten Speichers abzuspeichern, speichert man in der Zone 20 eine Adresse ab, die mit der Speicherung in einem zusätzlichen Teil 23 des Programmspeichers 15 (oder einem anderen Speicher des Chips 2) eines Befehls eines zusätzlichen Satzes zusammenhängt. Dieser Modus wird bevorzugt, da es ausreicht, durch den Server 1 lediglich die Adresse in dem zusätzlichen Teil 23 des Programmspeichers 15 von den anderen auszuführenden Befehlen: BILAN, ALGO, CHANGE, INVAL, ... verschicken zu lassen. Das ist kürzer. Fig. 2 zeigt den Ablauf des bevorzugten Automatismus zur Ausführung des anderen Befehls nach seiner Auswahl oder seinem Einspeichern. In dieser Figur sind die gleichen Elemente mit den gleichen Bezugszeichen wie in Fig. 1 versehen. Die Erfindung ist insbesondere für den Fall interessant, wo man mit dem Paar Schnittstelle 4, Chip 2 einen Befehl ausführen

lassen will, der nicht bereits in dem eingeschränkten Befehlssatz 11 oder 16 vorhanden ist. In der folgenden Beschreibung wird nicht berücksichtigt, ob es sich um eine serielle oder parallele Übertragung der Information handelt. In der Tat erfolgen die notwendigen Anpassungen durch den MODEM 8 einerseits und bei den bekannten Verwaltungsprotokollen der Speicher der Chipkarte durch die Lesegeräte 12 andererseits.

Eine Mitteilung 24 wird durch den Server 1 ausgegeben. Sie umfaßt im wesentlichen einen ersten Teil 25, der mit einem Befehlscode zusammenhängt, einen zweiten Teil 26, der mit einer designierten Speicherzone zusammenhängt (die, in der sich der Befehl befindet), und einen Teil 27, der mit einer Dateneinheit zusammenhängt. Der Teil 27 kann eine feste oder variable Länge haben und in diesem Fall bekanntermaßen vorne binär ein Byte haben, das die Länge der übertragenen Information angibt.

In einem ersten Fall 24 wird der Befehl ein Befehl SELECT sein, die betroffene Speicherzone wird der reservierte Speicher 20 sein, dessen Adresse MEM RES ist, und die Dateneinheit wird mit den Eigenschaften eines Befehls, z.B. ALGO, zusammenhängen. Der Mikroprozessor 9 der Schnittstelle 4 empfängt die Meldung 24 und überträgt jeweils auf seinem Befehlsbus 28, Adreßbus 29 und Datenbus 30 den in der Zone 25 enthaltenen Befehl, die in Zone 26 enthaltene Adresse und die in Zone 27 enthaltene Dateneinheit. Diese Übertragung zum Mikroprozessor 14 erfolgt gemäß den bekannten Protokollen und über das Lesegerät 12 und den Kontakt 18. Der Mikroprozessor 14 führt also den Befehl SELECT für die Meldung 24 oder UPDATE für eine Meldung 31 aus, indem

er mit dem Laden dieses Befehls in sein Befehlsregister 32 beginnt. Der betreffende Befehl wird anschließend durch den Mikroprozessor 14 ausgeführt, der den reservierten Speicher 20 auswählt bzw. aktualisiert. Zu diesem Zweck wird durch den Mikroprozessor 14 die Auswahladresse MEM RES über den Adreßbus 33 übertragen. Ein Befehl zur Ausführung des Befehls SELECT oder UPDATE wird über einen Befehlsbus 34 an einen Schaltkreis 35 zur Verwaltung des Speicher 20 angelegt. Der Schaltkreis 25 ist einfach ein Lese/Schreibschaltkreis, der zum Lesen oder Schreiben von über den Datenbus 36 übertragenen Daten im Speicher 20 an den über den Bus 33 übertragenen Adressen dient. Der Datenbus 36 überträgt die Daten, die vom Bus 33 empfangen wurden. Im Fall der Meldung 31 läßt der Mikroprozessor 14 durch den Schaltkreis 35 eine Einspeicherung ausführen, ein Schreiben in den Speicher 20 der Meldung ALGO, die vom Server 1 empfangen wurde. Man stellt hier fest, daß die Tatsache des Schreibens im reservierten Speicher 20 keinen Unterschied zum Schreiben in einem anderen Teil des Speichers des Chips 2 aufweist. Dieser Befehlstyp ist trivial. Der Automatismus der Erfindung hat also zum Ziel, in das Befehlsregister 32 des Mikroprozessors 14 den anderen Befehl zu übertragen, dessen Eigenschaften man soeben im reservierten Speicher 20 abgespeichert hat.

In der Praxis hat man in einer bevorzugten Ausführungsform im Speicher 20 nicht den Befehl selbst sondern eine Adresse des Befehls in dem zusätzlichen Satz 23 abgespeichert. Ebenso nach Einspeichern der Eigenschaften das gleiche bei der Adresse der bezeichneten zusätzlichen Instruktion: ALGO, man benutzt den Inhalt des Speichers 20, um die Adresse anzugeben und auf einen der Befehle im Programm-

speicher 23 zu zeigen. Mit anderen Worten liest beim bevorzugten Automatismus der Mikroprozessor 14 den Speicher 20 aus und überträgt die Dateneinheit, die gelesen wurde, auf seinem Adreßbus 33. Diese Adresse zeigt auf eine Speicherzone des Programmspeichers 23. Wenn einmal der Zeiger eingestellt ist, liest der Mikroprozessor also in der Speicherzone, auf die gezeigt wird, die Dateneinheit, die dort abgespeichert ist. Diese Dateneinheit ist in der Tat der andere Befehl. Der Mikroprozessor überträgt ihn also über seinen Befehlsbus 34 an sein Befehlsregister 32. Das Mikroprogramm des Automatismus endet mit der Ausführung der so geladenen anderen Instruktion.

Dieser andere Befehl kann zum Ziel haben, den Inhalt eines Speicheranhangs 21 des Chips 2 zu verändern. Er kann ebenso im invertierten Sinn an den Mikroprozessor 9 übertragen werden und an die Schnittstelle 4 angelegt werden für die Anzeige auf dem Schirm 13 oder ähnlich. Er kann ebenso direkt durch den Mikroprozessor 14 in bezug auf den Inhalt von einigen Speichern oder peripheren Einheiten der Schnittstelle 4 ausgeführt werden. Unter Berücksichtigung der Wahl des jeweiligen bevorzugten seriellen und parallelen Protokolls für eine Chipkarte 3 und eine Schnittstelle 4 wird man jedoch die Ausführung des anderen Befehls an den Mikroprozessor 9 übertragen. Dieser ist also in einem Befehlsregister des Mikroprozessors 9 in derselben Art wie beim Mikroprozessor 14 geladen.

Fig. 3 zeigt einen Ablauf, der für ein voreingetragenes Mikroprogramm in einem Programmspeicher 16 einer Chipkarte für das erfindungsgemäße Kommunikationssystem bevorzugt wird. Zu Beginn des Mikroprogramms erwartet der Mikropro-

zessor 14 die Eingabe eines Befehls. Anschließend liest er den Befehl ein, der über den Bus 28 übertragen wurde. Zu einem ersten Zeitpunkt verifiziert er im Verlauf eines Schrittes 37, daß dieser Befehl ein Auswahlbefehl ist. Wenn dies der Fall ist, führt er die Auswahl der bezeichneten Speicherzone, des Speichers 20 oder des Speichers 21, aus. Wenn dies nicht der Fall ist, testet der Mikroprozessor im Verlauf des Schrittes 38, ob der Befehl ein Lesebefehl READ ist. Wenn dies der Fall ist, wird das Lesen der Speicherzone veranlaßt, deren Adresse in Teil 26 der Meldung vereinbart wurde. Wenn dies nicht der Fall ist, wird im Schritt 39 untersucht, ob es sich um einen Aktualisierungsbefehl handelt und ggf. wird die betreffende Speicherzone aktualisiert. Schließlich muß normalerweise kein weiterer Test vorgesehen werden, und der verbliebene Befehl müßte ein Befehl STATUS sein, da die a priori die einzigen Befehle des beschränkten Befehlssatzes sind, die der Mikroprozessor erwarten kann.

Man stellt fest, daß man einen größeren oder evtl. kleineren beschränkten Satz vorsehen kann. Nur die Befehle SELECT und UPDATE sind wirklich für die Erfindung unverzichtbar. Ebenso verifiziert der Mikroprozessor in einem Schritt 40, ob der empfangene Befehl eine Aufforderung zur Ausführung von STATUS ist, und führt ihn ggf. aus. Es ist festzuhalten, daß der Befehl STATUS ein normalisierter Befehl sein könnte und nicht a priori die Bezeichnung der Speicherzone in der Zone 26 der Meldung beinhalten müßte.

Wenn im Gegenteil der Mikroprozessor 14 im Verlauf der Tests 37 bis 40 keinen erwarteten Befehl erkannt hat, liefert er in der Operation 41 eine Fehlermeldung aus (z.B.,

aber nicht notwendigerweise, darstellbar auf dem Schirm 13). Im Verlauf der Operation 42, die der Ausführung jedes der Befehle SELECT, READ, UPDATE und evtl. STATUS folgt, sucht der Mikroprozessor 14 herauszufinden, ob die in dem Teil 26 der Meldung enthaltene Adresse eine Adresse MEM RES des reservierten Speichers ist. Wenn dies nicht der Fall ist, betrachtet der Mikroprozessor 14 dies so, als ob die Behandlung der Meldung erledigt ist. Er begibt sich wieder in die Warteposition. Soweit es dem Befehl STATUS betrifft kann eine systematische Rückkehr in den Wartezustand auf Befehle des Mikroprozessors 14 vorgesehen werden, oder ebenso ein Durchlaufen des Tests 42, wenn die Zone 26 der Meldung gesetzt worden ist.

In dem Fall, wo die Zone 20 des reservierten Speichers betroffen ist, lädt der Mikroprozessor 14 den Inhalt dieses Speichers auf den Adressenbus 33. Der Adreßbus 33 zeigt also auf eine von mehreren Adressen des zusätzlichen Speichers 23. Je nachdem ob auf die erste, die zweite oder eine andere Adresse des Speichers 23 gezeigt wird, kann man mit einer Testreihe 43, 44 nacheinander die Auswahl im Speicher 23 eines unter mehreren Befehlen bewirken. Die Befehle dienen zum Speichern im Befehlsregister 32 des Mikroprozessors 14. Vorzugsweise wird der Speicher 23 ein Speicher vom nichtflüchtigen Typ mit Speicherzellen sein, die Transistoren mit offenem Gitter haben, vorzugsweise vom elektrisch beschreibbaren und löschbaren Typ: EEPROM. In diesem Fall wird nach Ausführung des zusätzlichen Befehls durch den Mikroprozessor 14 das Löschen des reservierten Speichers veranlaßt, so daß dieser wieder wie neu ist. Der Löschschritt 45 wird also nach der Ausführung des zusätzlichen Befehls durchgeführt.

Man kann jedoch auch anders vorgehen. Man kann z.B. vorsehen, daß der Aktualisierungsbefehl UPDATE, der vom Mikroprozessor 14 (nicht vom Mikroprozessor 9) durchgeführt wird, selbst ein Mikroprogramm enthält, das darin besteht, vorläufig den reservierten Speicher 20 vor dem Einschreiben eines neuen Zeigers in Hinsicht auf die Ausführung eines anderen zusätzlichen Befehls zu löschen. Dies ist von Vorteil, wenn die Einheitenabrechnung in der Speicherzone 21 der Chipkarte 3 veranlaßt werden muß. In der Tat schickt in diesem Fall der Server 1 das erste Mal einen Befehl UPDATE, um einen Befehl zur Abrechnung der Einheiten in der zusätzlichen Zone 23 zu markieren. In der Folge genügt es für ihn, den reservierten Speicher auszuwählen, um die Abrechnung der Einheiten automatisch auszuführen. In der Tat ist der Befehl zur Abrechnung bereits im Speicher 20 eingeschrieben, es ist nicht notwendig, ihn erneut zu schreiben. Es reicht, ihn auszuwählen, zu lesen und den Schritt 42 zu durchlaufen. Bei Bedarf kann selbst der Befehl STATUS verwendet werden. Die Meldung 31 umfaßt im Unterschied zur Meldung 24 in Zone 27 weitere Daten. Diese Daten zielen auch auf den reservierten Speicher, sie können in unterschiedlichen Speicherzonen 46 oder 47 des reservierten Speichers gespeichert sein. In diesem Fall wird ein erster Teil des Inhalts der Meldung in Zone 27 in einer ersten Zone 46 des Speichers 20 gespeichert werden, ein zweiter Teil in einer anschließenden Zone 47 usw. Dies bedeutet, daß vor dem Übertragen nach und nach alle Befehle, um sie durch den Mikroprozessor ausführen zu lassen, der Server 1 eine einzige Meldung abschickt, die ein Folge von Befehlen vor der sequentiellen Ausführung umfaßt.

In diesem Fall umfaßt um ein Beispiel zu geben der Speicher 20 in jeder Zone ein Feld 48, worin die Zeiger oder Eigenschaften der auszuführenden Befehle gespeichert sind, und ein Feld 49, das z.B. ein Bit ist oder ein anderes System, in welchem eine Binärinformation bedeutet, daß nach Ausführung eines Befehls des zusätzlichen Satzes bei Anwesenheit einer Eins die Ausführung eines anderen Befehls des zusätzlichen Satzes folgen muß. Demgegenüber folgt aus der Anwesenheit einer Null, daß keine weiteren auszuführenden Befehle folgen. Zu diesem Zweck wird die Ausführung jedes Befehls des zusätzlichen Satzes in dem Mikroprogramm in Fig. 4 einen Test 50 umfassen, in dessen Verlauf man herauszufinden sucht, ob der Ausführung eines zusätzlichen Befehls die Ausführung eines anderen folgenden zusätzlichen Befehls folgen muß oder nicht. Die Verzweigung des Mikroprogramms muß also unmittelbar sein.

In einer Variante verwendet man eine existierende Prozedur zur Verwaltung der Karte. Zum Beispiel umfaßt eine existierende Prozedur zur Anerkennung der Karte das Abschicken einer chiffrierten Zufallsfolge durch den Server an die Karte, die Dechiffrierung dieser Zufallsfolge durch die Karte, die Verschlüsselung durch einen Algorithmus vom Typ DES, parametrisiert durch die Zufallsfolge, die durch einen geheimen Code dechiffriert wurde, und das Verschicken an das Lesegerät zur Verifizierung des verschlüsselten geheimen Codes. Bei der Erfindung reserviert man eine gegebene Zufallsfolge, z.B. 0000XXXX, um einerseits durch die 0000 anzuzeigen, daß man es nicht mit einer wirklich existierenden Prozedur zu tun hat, und andererseits die Daten XXXX in den Speicher 20 zu laden. Der Rest wird auf die oben beschriebene Art ausgeführt. Der Vorteil, eine existierende Proze-

dur zu verwenden, liegt darin, daß die existierenden (zahlreichen) Schnittstellen bereits kompatibel mit dieser Prozedur sind. Es reicht, den Test, ob 0000 vorliegt, in das Betriebssystem von neu hergestellten Karten einzubauen.

Ansprüche

1. Kommunikationssystem, das umfaßt:
einen Zentralserver (1), einen elektronischen Chip (2) auf einem Chipträger (3) und eine Schnittstelle (4) für die Kommunikation zwischen Zentralserver, diesem Chip und eventuell einem Nutzer,
in der Schnittstelle einen Mikroprozessor (9) und einen Programmspeicher (11), der mit einem begrenzten Satz (11) an Instruktionen oder Prozeduren zur Kommunikation mit dem Chip ausgerüstet ist, und
in dem Chip einen Mikroprozessor (14) und einen Programmspeicher (15), der gleichfalls mit einem entsprechenden begrenzten Satz (16) an Instruktionen oder Prozeduren ausgestattet ist,
dadurch g e k e n n z e i c h n e t , daß das Kommunikationssystem umfaßt:
Vorrichtungen, die dazu dienen, aufgrund einer vom Zentralserver ausgegebenen Meldung im Verlauf einer Nutzungsperiode in einer reservierten Speicherzone (20) des Chips mit den begrenzten Sätzen an Instruktionen oder Prozeduren des Chips und/oder der Schnittstelle die Eigenschaften einer Instruktion auszuwählen oder zu schreiben, die sich von denen der begrenzten Sätze oder Prozeduren unterscheidet, und
in dem Chip einen Ausführungsautomatismus, der dazu dient, im Verlauf dieser Periode diese unterschiedliche Instruktion nach Auswahl oder nach Schreiben ihrer Eigenschaften in diese reservierte Speicherzone auszuführen.

2. System nach Anspruch 1,
dadurch gekennzeichnet, daß
der Programmspeicher des Chips einen zusätzlichen Instruktionsatz umfaßt,
die geschriebenen Eigenschaften eine Adresse in diesem Programmspeicher des Chips einer Instruktion dieses zusätzlichen Satzes betreffen.

3. System nach Anspruch 1 oder Anspruch 2,
dadurch gekennzeichnet, daß
der Automatismus im Programmspeicher des Chips umfaßt:
ein Mikroprogramm zum Testen (37 - 40), ob eine Instruktion, die vom Server oder Nutzer empfangen wurde, zum begrenzten Instruktionssatz gehört, der an die Ausführung dieser getesteten Instruktion gekoppelt ist, in Serie geschaltet mit
einem Mikroprogramm zum Testen (42) der Bezeichnung, in der so ausgeführten Instruktion, der reservierten Speicherzone, gekoppelt mit der Ausführung der sich unterscheidenden Instruktion.

4. System nach Anspruch 1 oder Anspruch 2,
dadurch gekennzeichnet, daß
der Automatismus im Programmspeicher des Chips umfaßt:
ein Mikroprogramm zum Testen (42) der Bezeichnung der reservierten Speicherzone, gekoppelt mit der Ausführung der sich unterscheidenden Instruktion.

5. System nach Anspruch 3 oder Anspruch 4,
dadurch gekennzeichnet, daß
der Automatismus im Programmspeicher des Chips umfaßt:

ein Mikroprogramm zur Ausführung einer Instruktion des zusätzlichen Satzes, deren Adresse in der reservierten Speicherzone gespeichert ist.

6. System nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, daß die reservierte Speicherzone des Chips nichtflüchtig ist, insbesondere ein EEPROM-Typ ist.

7. System nach einem der Ansprüche 1 bis 6, dadurch gekennzeichnet, daß die reservierte Speicherzone ein Feld (49) zum Speichern einer Information umfaßt, wodurch es möglich wird, festzustellen, ob nach Ausführung einer sich gebenüber denen aus dem begrenztem Satz oder den Prozeduren unterscheidenden Instruktion die Ausführung einer anderen sich unterscheidenden Instruktion folgen muß, und dadurch daß der Ausführungsautomatismus einen Test (50) des Wertes dieses Feldes umfaßt.

8. System nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, daß der Mikroprozessor der Schnittstelle zur Ausführung der sich unterscheidenden Instruktion dient.

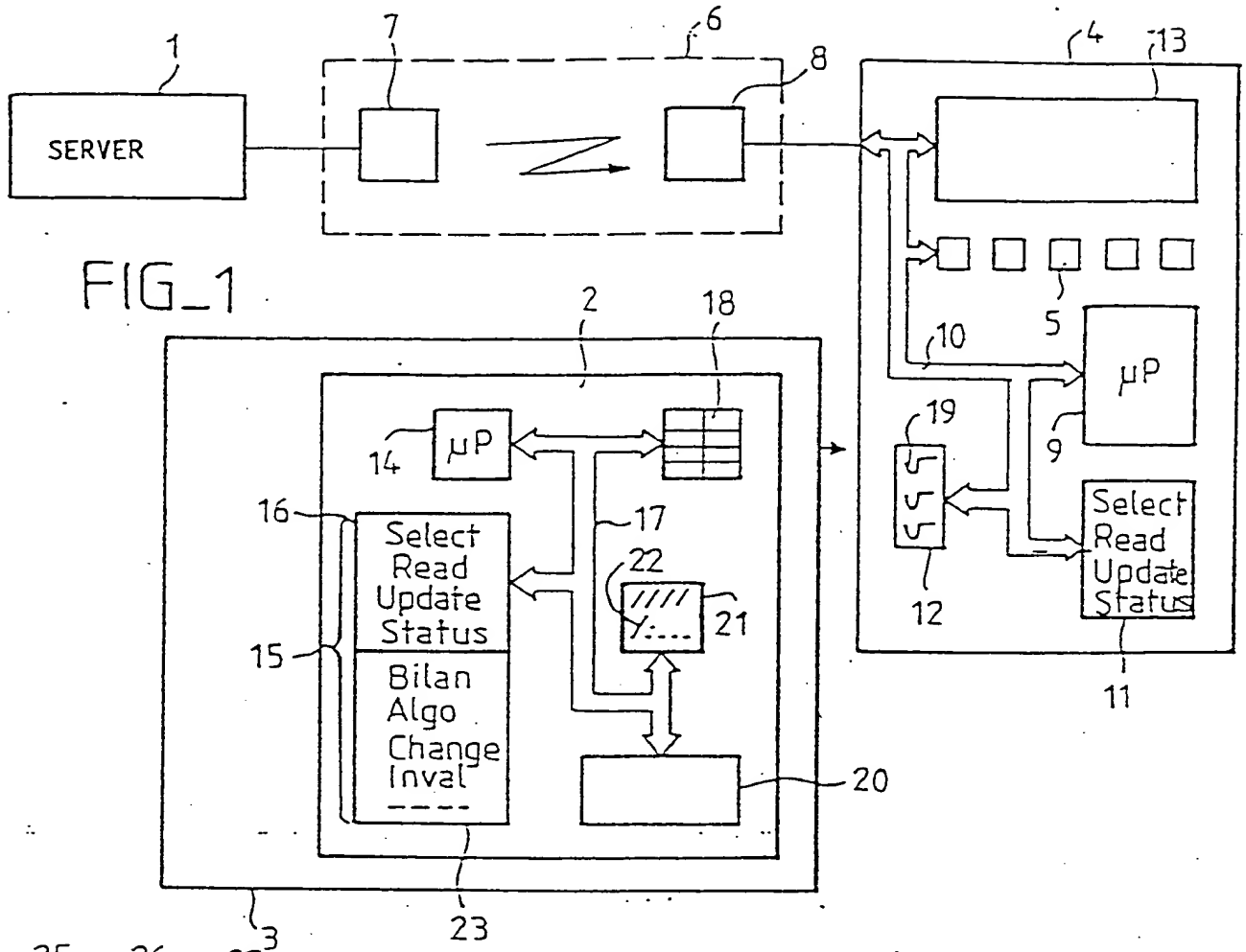

SERVER
1
7
6
8
4
13
5
μP
10
9
19
12
Select
Read
Update
Status
11
2
18
14
μP
16
Select
Read
Update
Status
15
Bilan
Algo
Change
Inval
17
22
21
20
23
3

FIG_1

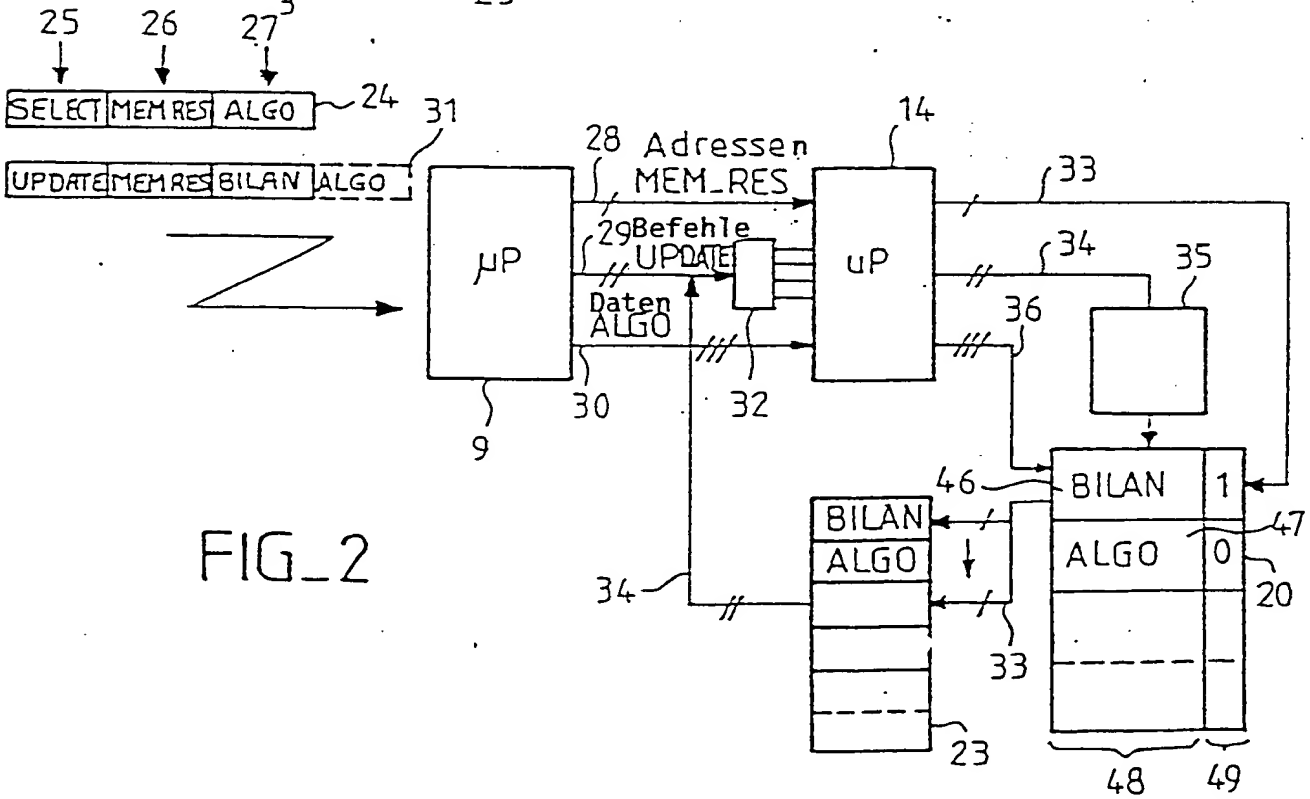
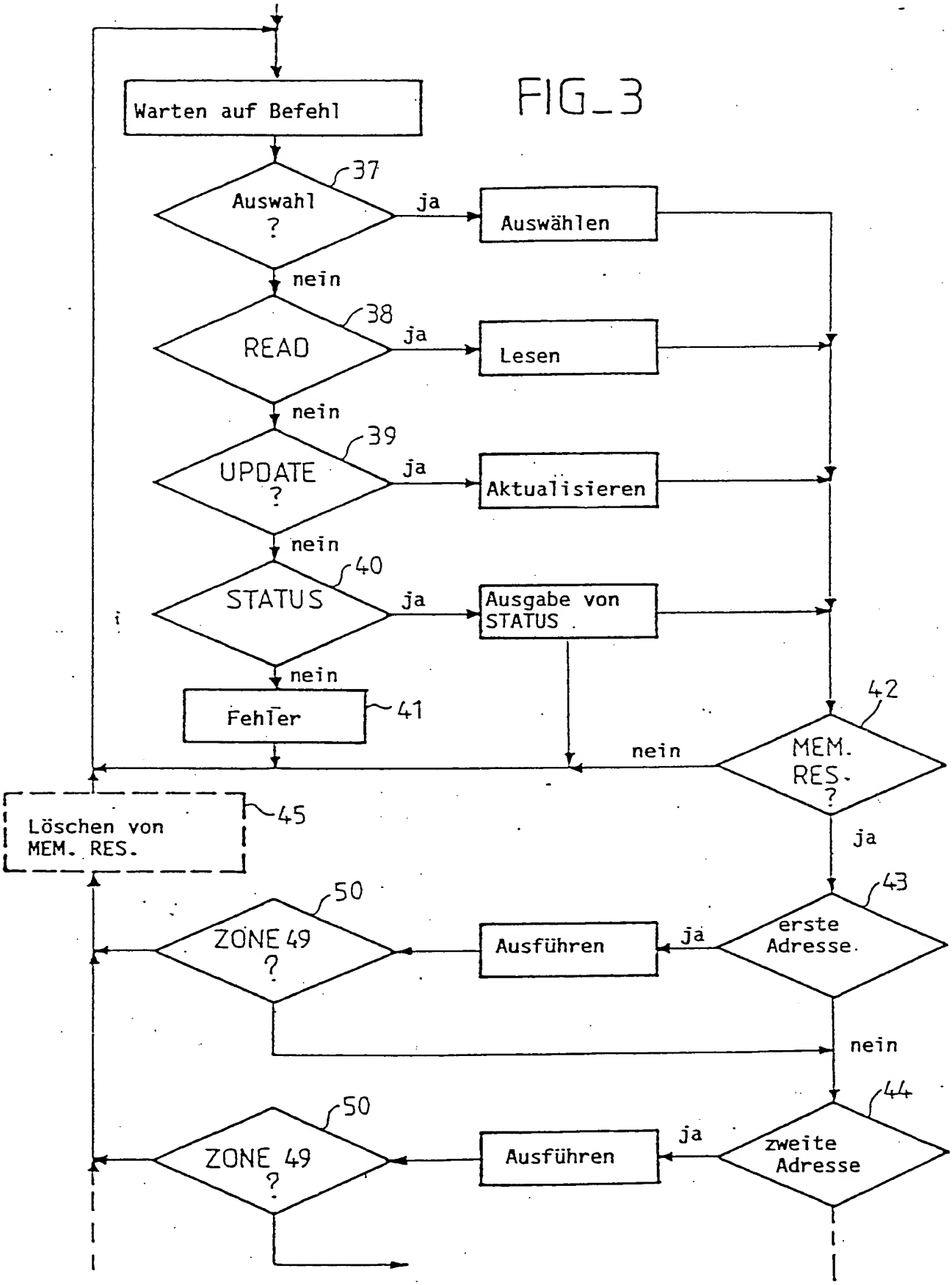

25
26
27
SELECT
MEM RES
ALGO
24
31
UPDATE
MEM RES
BILAN
ALGO
μP
9
28 Adressen
MEM_RES
29 Befehle
UPDATE
Daten
ALGO
30
32
14
uP
33
34
35
36
BILAN
1
ALGO
0
47
20
46
BILAN
ALGO
34
33
23
48
49

FIG_2

FIG_3


Warten auf Befehl
37
Auswahl ?
ja
Auswählen
nein
38
READ
ja
Lesen
nein
39
UPDATE ?
ja
Aktualisieren
nein
40
STATUS
ja
Ausgabe von STATUS
nein
Fehler
41
42
MEM. RES. ?
nein
Löschen von MEM. RES.
45
ja
43
erste Adresse
ja
Ausführen
50
ZONE 49 ?
nein
44
zweite Adresse
ja
Ausführen
50
ZONE 49 ?